

A First Course on Kinetics and Reaction Engineering

Example 26.3

Problem Purpose

This problem will help you determine whether you have mastered the learning objectives for this unit.

Problem Statement

A perfectly insulated tubular reactor with a diameter of 10 cm and a length of 5 m is fed an aqueous solution containing A and B at concentrations of 1.0 and 1.2 M, respectively. This feed stream is at a constant temperature of 30 °C and flows at 75 L min⁻¹. Reagents A and B react according to reaction (1) with a rate of reaction that is accurately described by equation (2). The heat of reaction (1) is -10,700 cal mol⁻¹ and may be assumed to be constant. The heat capacity of the solution and the density of the solution may be taken to be constant and equal to those of water (1.0 cal g⁻¹ K⁻¹ and 1.0 g cm⁻³). The pressure drop in the reactor is negligible. Plot the steady state molar flow rate of A and the temperature as a function of distance into the reactor.



$$r_1 = \left(8.72 \times 10^5 \text{ L mol}^{-1} \text{ min}^{-1}\right) \exp\left\{\frac{-7200 \text{ cal mol}^{-1}}{RT}\right\} C_A C_B \quad (2)$$

Problem Analysis

The reactor in this problem is a steady state PFR operating adiabatically. The reaction kinetics are known; this is a reaction engineering question. To answer this question, mole balance design equations will be written for each reactant and product in the system. An energy balance will also be written, but a mechanical energy balance is not needed since the problem states that the pressure drop is negligible. The design equations will then be solved for a variety of reactor lengths between zero and 5 m. Each solution will yield the molar flow rate of A and the temperature at the corresponding value of z , the distance into the reactor. The results can then be plotted to obtain the requested solution.

Problem Solution

The following quantities are provided in the problem statement: $D = 10 \text{ cm}$, $L = 5 \text{ m}$, $C_A^0 = 1.0 \text{ mol L}^{-1}$, $C_B^0 = 1.2 \text{ mol L}^{-1}$, $\dot{n}_Y^0 = 0$, $\dot{n}_Z^0 = 0$, $T^0 = 30 \text{ °C}$, $k_0 = 8.72 \times 10^5 \text{ L mol}^{-1} \text{ min}^{-1}$, $E = 7.2 \text{ kcal mol}^{-1}$, $\Delta H(T) = -10700 \text{ cal mol}^{-1}$, $\tilde{C}_p = 1.0 \text{ cal g}^{-1} \text{ K}^{-1}$, $\rho = 1.0 \text{ g cm}^{-3}$ and $\dot{V}^0 = 75 \text{ L min}^{-1}$. The system can be represented schematically as shown in Figure 1. In constructing Figure 1, it was noted that since the density of the fluid is constant, the outlet volumetric flow rate will equal the inlet volumetric flow rate. In addition, it was noted that the inlet molar flow rates of A and B are constant and equal to the product of

the inlet concentration and the inlet volumetric flow rate. Since the problem does not mention the presence of Y or Z in the feed, the inlet molar flow rates of Y and Z are inferred to equal zero.

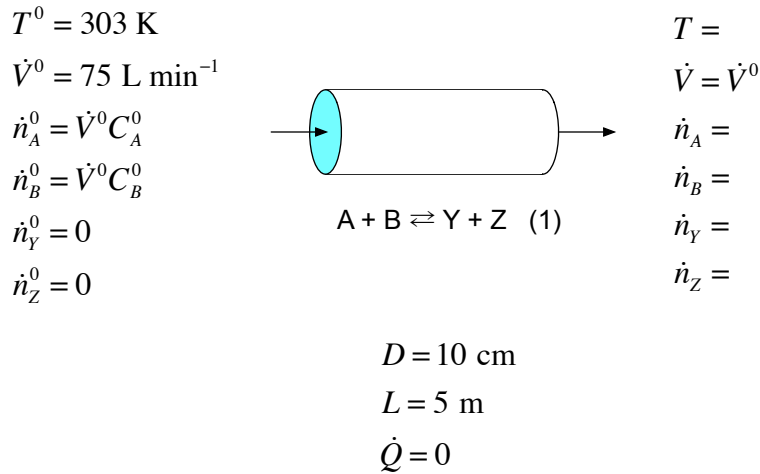


Figure 1. Schematic representation of the reactor.

The generalized, steady-state mole balance equation for a PFR is given in equation (3). Since only one reaction is taking place, the summation will reduce to a single term. Equation (3) can be used to generate the mole balances for each of the reactants and products in the system as given in equations (4) through (7). Note, one could also write a mole balance on the solvent, water, but we will see here that it is not needed in order to solve the problem. Equations (4) through (7) are each in the standard form with a single derivative on the left of the equals sign and a function of the independent and dependent variables on the right of the equals sign.

$$\frac{d\dot{n}_i}{dz} = \frac{\pi D^2}{4} \left(\sum_{j=1}^{N_{\text{rxns}}} \nu_{i,j} r_j \right) \quad (3)$$

$$\frac{d\dot{n}_A}{dz} = f_1(z, \dot{n}_A, \dot{n}_B, \dot{n}_Y, \dot{n}_Z, T) = -\frac{\pi D^2}{4} r_1 \quad (4)$$

$$\frac{d\dot{n}_B}{dz} = f_2(z, \dot{n}_A, \dot{n}_B, \dot{n}_Y, \dot{n}_Z, T) = -\frac{\pi D^2}{4} r_1 \quad (5)$$

$$\frac{d\dot{n}_Y}{dz} = f_3(z, \dot{n}_A, \dot{n}_B, \dot{n}_Y, \dot{n}_Z, T) = \frac{\pi D^2}{4} r_1 \quad (6)$$

$$\frac{d\dot{n}_Z}{dz} = f_4(z, \dot{n}_A, \dot{n}_B, \dot{n}_Y, \dot{n}_Z, T) = \frac{\pi D^2}{4} r_1 \quad (7)$$

The general steady-state energy balance equation for a PFR is given in equation (8), which for this problem simplifies to equation (9). In simplifying equation (8) the summation in the denominator was replaced by the heat capacity of the solution as a whole, the heat transfer term in the numerator was

eliminated since the reactor is adiabatic, and the summation in the numerator was expanded. Equation (9) is also in the standard form. Since pressure drop can be neglected, a mechanical energy balance is not needed.

$$\frac{dT}{dz} = \frac{\pi DU(T_e - T) - \frac{\pi D^2}{4} \left(\sum_{j=1}^{N_{rxns}} r_j \Delta H_j \right)}{\left(\sum_{i=1}^{N_{species}} \dot{n}_i \hat{C}_{p,i} \right)} \quad (8)$$

$$\frac{dT}{dz} = f_5(\dot{n}_A, \dot{n}_B, \dot{n}_Y, \dot{n}_Z, T) = \frac{-\frac{\pi D^2}{4} r_1 \Delta H_1^0(T)}{\dot{V} \rho_{fluid} \tilde{C}_{p,fluid}} \quad (9)$$

Design equations (4) through (7) and (9) represent a set of 5 ordinary differential equations (ODEs) with 5 dependent variables (\dot{n}_A , \dot{n}_B , \dot{n}_Y , \dot{n}_Z and T) and one independent variable (z). They are initial value ODEs because the value of each of the 5 dependent variables is known at the same value ($z = z_0 = 0$) of the independent variable. A set of n initial value ODEs containing n dependent variables and one independent variable can be solved numerically to obtain the values of all of the variables at some other point of interest. If you are not familiar with the numerical solution of initial value ODEs, Supplemental Unit S5 of “A First Course on Kinetics and Reaction Engineering” offers a brief introduction to the topic and provides MATLAB template files named SolvIVDifI.m and SolvIVDifD.m that can be used as a starting point for solving equations of this kind.

Generally, no matter what software package is used, the numerical solution of these equations will require you to provide (a) the initial values (z_0 , \dot{n}_A^0 , \dot{n}_B^0 , \dot{n}_Y^0 , \dot{n}_Z^0 and T^0), (b) the final value of either the independent variable or one of the dependent variables at the point of interest and (c) code that, given values for the independent and dependent variables, evaluates the functions f_1 through f_5 in equations (4) through (7) and (9). The initial values (at $z = z_0 = 0$) needed to solve the design equations (\dot{n}_A^0 , \dot{n}_B^0 , \dot{n}_Y^0 , \dot{n}_Z^0 and T^0) are given in Figure 1. The final value is that the reactor is 5 m long ($z = z_f = L = 5$ m).

Looking at the functions f_1 through f_5 , the only quantity other than the independent and dependent variables that is not known is r_1 . The code that will evaluate functions f_1 through f_5 will be given values of z , \dot{n}_A , \dot{n}_B , \dot{n}_Y , \dot{n}_Z and T , so all it will need to calculate is the rate. This can be computed using equation (2), but in order to do so, concentrations are needed. By definition, these can be computed using equations (10) and (11).

$$C_A = \frac{\dot{n}_A}{\dot{V}} \quad (10)$$

$$C_B = \frac{\dot{n}_B}{\dot{V}} \quad (11)$$

In this problem we are asked to plot the molar flow rate of A and the temperature along the length of the reactor. To do so, the design equations are integrated from $z = 0$ (the inlet) to varying final values of z that span the range from zero to the given reactor length, L . Doing so for any one final value of z yields the values of the dependent variables at that axial position. Since the molar flow rate of A and the temperature are dependent variables, no further calculations are needed. The resulting data can then be plotted, yielding Figures 2 and 3.

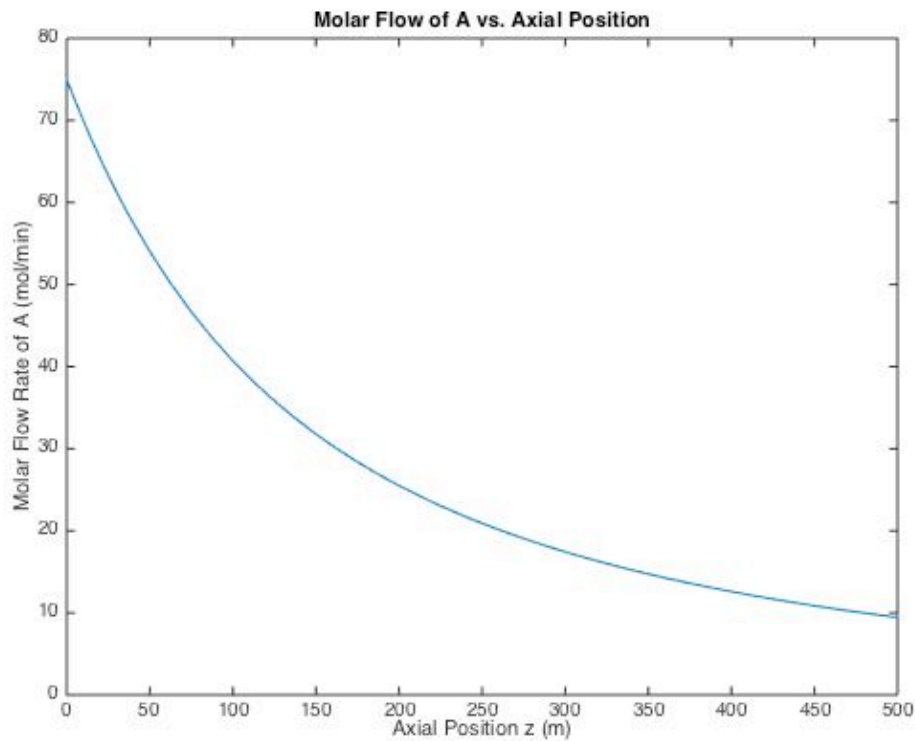


Figure 2. Molar flow rate of A as a function of axial position in the reactor.

The curve in Figure 2 shows the shape one would expect on the basis of a qualitative analysis of the PFR, as discussed in Unit 25. It starts at its maximum value and then decreases, with a decreasing slope until it eventually asymptotically approaches the equilibrium value. In this case, the reaction is irreversible, so if the reactor was longer, the molar flow rate of A would eventually become equal to zero. Similarly, the temperature displays the expected behavior by steadily increasing at a decreasing rate until it asymptotically reaches a constant value.

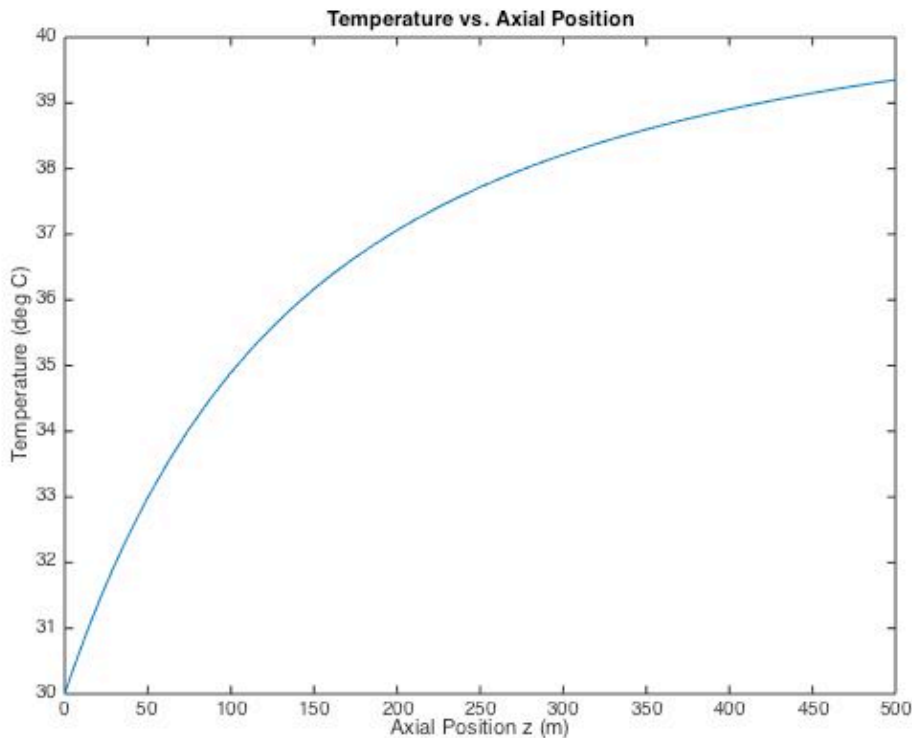


Figure 3. Temperature as a function of axial position in the reactor.

Calculation Details Using MATLAB

If you elect to use MATLAB to solve the design equations, Supplemental Unit S5 provides template files that can be used. In this problem, the equations are initial value ODEs and the final value of the independent variable is provided, so the appropriate template file is SolvIVDifl.m. Before that file can be used, you must make four required modifications. Here I will also describe a few non-required modifications that you might want to consider when solving problems of this type. In particular, you might suspect that, like in Example 26.2, it will be necessary to create some kind of loop that repeats the calculations many times using a different value of L each time and saving the results for subsequent plotting. We will see that when using MATLAB this is not necessary.

To begin, I made a copy of the template file and saved it as Example_26_3.m; a copy of that file accompanies this solution. Since the function name must match the filename, I changed the name of the function to Example_26_3. At the same time, knowing that I won't need to use the results from these calculations in subsequent calculations, I changed the function so that it does not return any values. The template file begins with a long set of comments describing what it does and how to use it; I replaced these comments with a brief comment stating the purpose of the modified version. None of these modifications were required. The first *required* modification involves entering all the known quantities from the problem statement along with universal constants that will be needed (from handbooks or other

reference sources). As these were entered, they were converted to a consistent set of units. At the same time I calculated the inlet molar flow rates of A and B, since those quantities are also constants. The result of making all these modifications is shown in Listing 1.

```

% Modified version of the MATLAB template file SolvIVDifI.m used in the
% solution of Example 26.3 of "A First Course on Kinetics and Reaction
% Engineering."
%
function Example_26_3
    % Known quantities and constants
    D = 10; % cm
    L = 500; % cm
    CA0 = 1.0/1000.0; % mol/cm^3
    CB0 = 1.2/1000.0; % mol/cm^3
    nY0 = 0;
    nZ0 = 0;
    T0 = 30 + 273.15; % K
    k0 = 8.72E5*1000.; % cm^3/(mol min)
    E = 7200; % cal/mol
    dH = -10700; % cal/mol
    Cp = 1.0; % cal/(g K)
    rho = 1.0; % g/cm^3
    VFR = 75*1000; % cm^3
    % Universal constants
    R = 1.987; % cal/(mol K)
    % Calculated constants
    nA0 = CA0*VFR;
    nB0 = CB0*VFR;

```

Listing 1. Initial comment, function declaration and known constants after modification of SolvIVDifI.m

The second required modification involves entering code that, given values for the independent and dependent variables, evaluates the functions f_1 through f_5 in equations (4) through (7) and (9). This takes place within an internal function named `odeqns`. Notice that within that internal function, the equations are provided as a vector quantity. Thus, it is necessary to map the dependent variables used in the problem statement (\dot{n}_A , \dot{n}_B , \dot{n}_Y , \dot{n}_Z and T) to a vector z , and the corresponding derivatives are mapped to a vector $dzdt$. (Don't be confused: in the template file z represents the vector of dependent variables and t represents the independent variable whereas in our problem, z is the variable used for the independent variable. In other words, the independent variable, z , used in the problem statement maps to the variable t in the MATLAB code.) I find it useful at the start of the internal function that will evaluate the derivatives, to define local variables with the names used in the problem statement. This modification is not required, but in my opinion, it makes the code more readable and easier to debug. In addition, the resulting list of variables serves as a reminder of the mapping of the problem statement variables to the vector z . The required code first calculates the concentrations of A and B that appear in the rate expression according

to equations (10) and (11). Following that, the rate is calculated using equation (2). Finally, the functions f_1 through f_5 are evaluated using equations (4) through (7) and (9), saving the results in the vector $dzdt$.

The resulting code is shown in Listing 2.

```
% Function that evaluates the ODEs
function dzdt = odeqns(t,z)
    nA = z(1);
    nB = z(2);
    nY = z(3);
    nZ = z(4);
    T = z(5);
    CA = nA/VFR; % equation (10)
    CB = nB/VFR; % equation (11)
    r = k0*exp(-E/R/T)*CA*CB; % equation (2)
    dzdt = [
        -pi()*D^2/4*r % equation (4)
        -pi()*D^2/4*r % equation (5)
        pi()*D^2/4*r % equation (6)
        pi()*D^2/4*r % equation (7)
        -pi()*D^2/4*r*dH/VFR/rho/Cp % equation (9)
    ];
end % of internal function odeqns
```

Listing 2. Results of the second required modification.

The third required modification involves providing the initial values of the independent and dependent variables and the final value of the independent variable. Recall that the independent variable, z , from the problem statement maps to the variable t in the code. Thus t_0 in the code is set equal to the initial value of the axial position, which is denoted as $z_0 = 0$ in the problem statement. The initial values of the dependent variables are entered as a vector named z_0 , and they must use the same mapping of the problem variables (\dot{n}_A^0 , \dot{n}_B^0 , \dot{n}_Y^0 , \dot{n}_Z^0 and T^0) to the vector z_0 as was used previously to map (\dot{n}_A , \dot{n}_B , \dot{n}_Y , \dot{n}_Z and T) to z . As with the initial value, the final value, z_f , in the problem statement, maps to t_f in the code. The results of performing this modification are shown in Listing 3.

```
% Initial and final values
t0 = 0;
z0 = [
    nA0
    nB0
    nY0
    nZ0
    T0
];
tf = L;
```

Listing 3. Results of the third required modification.

The fourth and final *required* modification is to use the results from solving the ODEs to calculate whatever the problem requested. In this case all that is necessary is to plot the results. Remember earlier I said that we did not need to use a loop to calculate and save values for plotting? As noted in Supplemental Unit S5, the way an initial value ODE solver in MATLAB works is to increase the independent variable in small steps, solving the ODEs at each step. When it reaches the final condition, it stops and returns the value of the independent variable and the corresponding values of the dependent variables at each of those steps. In other words, it returns all the data we need to make the plot.

In the template file, the names of the returned variables are `t` and `zz`. The vector, `t`, contains the starting value of the independent variable and its value at each of the steps the solver took. The matrix, `zz`, has one column for each dependent variable. The rows contain the values of the dependent variables corresponding to each of the values of the independent variable in the vector, `t`. Thus, to make the requested plots, all we need to do is plot the appropriate column of the matrix, `zz`, versus the vector, `t`. The mapping in Listing 2 shows that `z(1)` corresponds to \dot{n}_A and `z(5)` corresponds to the temperature; therefore the requested plot of \dot{n}_A vs. axial position (z) is a plot of the first column of the matrix `zz` versus the vector `t` (which maps to the axial position). The requested plot of T vs. axial position (z) is a plot of the fifth column of the matrix `zz` versus the vector `t` (which maps to the axial position). The code shown in Listing 4.

```
% Generate the plots
figure
plot(t,zz(:,1))
title('Molar Flow of A vs. Axial Position')
xlabel('Axial Position z (m)')
ylabel('Molar Flow Rate of A (mol/min)')
figure
plot(t,zz(:,5)-273.15)
title('Temperature vs. Axial Position')
xlabel('Axial Position z (m)')
ylabel('Temperature (deg C)')
```

Listing 4. Results of the fourth required modification.

Once the file containing all the modifications had been saved, it was executed by typing `Example_26_3` at the MATLAB command prompt. Doing so generated Figures 2 and 3, shown previously.